Name: Feifan Liao
ID:fl2656

Problem 1c: Extra Credit

Answer:
the *_getitem_* method assume that "words" is a iterable variable (since: "words: Iterable[str]"), but if we use "embeddings["the"]", the input "words" value is "the", which is a string and will be treated as a iterable variable of ("t", "h", "e") instead of "the" as a whole.
Therefore, we will not get the embedding result of "the" (unlike embeddings["the", "of"], because the input is a tuple, which allows a proper embedding result of "the" and "of"). What's worse, "h" is not in our "glove_50d.txt" dataset, so the code will lead to a key error of "KeyError: 'h'".

Problem 4a: Syntactic vs. Semantic Relation Types

Answer:
The accuracy result obtained by each GloVe embedding space (50, 100 and 200) are shown as the table below. The results for the two word2vec models (Skip-Gram and CBOW) reported in Table 4 of Mikolov et al. (2013) are also shown in the table for comparison (in order to answer question 4a.4).

Method	Embedding Space	Semantic	Syntactic	Overall
Ours	GloVe 50	40.0%	27.6%	33.2%
Ours	GloVe 100	44.5%	27.8%	35.4%
Ours	GloVe 200	31.7%	21.7%	26.2%
CBOW	(Dim = 300)	15.5%	53.1%	36.1%
Skip-gram	(Dim = 300)	50.0%	55.9%	53.3%

1. What is the total analogy question accuracy obtained by each embedding space for all analogies belonging belonging to a **semantic** relation type?

(See the results as the table above)

2. What is the total analogy question accuracy obtained by each embedding space for all analogies belonging belonging to a **syntactic** relation type?

(See the results as the table above)

3. What is the total analogy question accuracy obtained by each embedding space for **all analogies** in the testing dataset?

(See the results as the table above)

4. How do your GloVe results compare to the results for the two word2vec models (Skip-Gram and CBOW) reported in Table 4 of Mikolov et al. (2013)? Does the dimensionality of the embedding space have any effect on analogy question accuracy?

On semantic task, the Skip-gram method outperforms all the other methods with 50.0% accuracy, while our method using GloVe dataset achieved 31.7% to 44.5% in this task (in which GloVe 100 get the best beformance), and the CBOW have the lowest accuracy of 15.5% (half than the second worst case, GloVe 200)

On syntactic analogy tasks, GloVe performs significantly worse than both Word2Vec models. CBOW and Skip-Gram, which respectively achieve 53.1% and 55.9% accuracy, whereas GloVe 50, 100 and 200 only reaches 27.6%, 27.8% and 21.7%.

In terms of overall accuracy, GloVe 100 performs best among GloVe-based methods and is similarly to CBOW (35.4% vs. 36.1%). However, it is far behind Skip-Gram (with the accuracy of 53.3%).

In terms of dimensionality, increasing GloVe's embedding size doesn't necessary increasing the accuracy in different tasks. To be specific, increasing GloVe from 50d to 100d does improve semantic accuracy (from 40.0% to 44.5%) but has little effect on syntactic tasks (from 27.6% to 27.8%). Increasing the dimensionality further to 200d even leads to reduction of performance (Semantic: from 44.5 to 31.7%, Syntactic: from 27.8% to 21.7%). Therefore, we should not blindly improving performance by increasing the dataset dimensionality.

Problem 4b: Effect of Lenience

Answer:
The accuracy result obtained by each GloVe embedding space (50, 100 and 200, using a lenience of 'k=1' and 'k = 2') are shown as the table below.

Embedding Space	**Semantic(k=1 -> k=2)**	**Syntactic(k=1 -> k=2)**	**Overall(k=1 -> k=2)**
GloVe 50	40.0% -> 56.6%	27.6% -> 53.6%	33.2% -> 55.0%
GloVe 100	44.5% -> 66.5%	27.8% -> 65.9%	35.4% -> 66.2%
GloVe 200	31.7% -> 70.5%	21.7% -> 67.2%	26.2% -> 68.7%

As the table above, the results change significantly when lenience is increased from k=1 to k=2.

In terms of improvement under all tasks and embedding spaces, all of their performance increased. The lowest increasement appear in the case of GloVe 50 in semantic task, which is 40.0% to 56.6% (+16.6%). And the GloVe 200 in syntactic task get the greatest increasement, which is 21.7% to 67.2% (+45.5%).

In terms of improvement under different tasks, the syntactic task benefits most from the change of lenience since the accuracy increase from 21.7% - 27.8% to 53.6% - 67.2%.

In terms of improvement under different embedding spaces, embeddings with higher dimension benfit more than others. Though we conclude that change GloVe 100 to GloVe 200 might result in no or even negative improvement when k=1, when we set k=2, the increase of embedding space dimension clearly improve the accuracy in semantic, syntactic and overall tasks.

Problem 4c: Qualitative Evaluation

Answer:
The analogy questions result obtained by each GloVe embedding space (50, 100 and 200) are shown as the table below. The results by Mikolov et al. are also shown in the table for comparison (in order to answer question 4c).

Analogy Question	**Gold Answer**	**GloVe 50**	**GloVe 100**	**GloVe 200**	**Skip-gram(By Mikolov et al.)**
france : paris :: italy : *x*	rome	rome	rome	rome	rome
france : paris :: japan : *x*	tokyo	tokyo	tokyo	tokyo	tokyo
france : paris :: florida : *x*	tallahassee	miami	florida	florida	Tallahassee
big : bigger :: small : *x*	smaller	larger	larger	smaller	larger
big : bigger :: cold : *x*	colder	cold	cold	cold	colder
big : bigger :: quick : *x*	quicker	quick	quick	quick	quicker

In terms of different GloVe embedding spaces, it seems that all GloVe spaces perform better in semantic task than sytactic tasks. While the bigger GloVe spaces seem to be able to get more reasonable prediction, because the answer of "miami" changes to "florida" in "france : paris :: florida : x" (still wrong, but more reasonable), and "larger" changes to "smaller" in "big : bigger :: small : x".

In comparsion with the result reported by Mikolov et al. (skip-garm), all GloVe spaces have a lower performance. The skip-garm successfully predict the gold answer of "Tallahassee", "colder" and "quicker", on which all trials by GloVe spaces are wrong.